A Scalable Communication Architecture for AMI in SmartGrid

Hoang Giang Ngo, Luigi Liquori, Chan Hung Nguyen

A Scalable Communication Architecture for AMI in SmartGrid

Giang Ngo Hoang, Luigi Liquori, Hung Nguyen Chan

RESEARCH REPORT
N° 8410
03 December 2013
Project-Teams Lognet

ISSN 0249-6399 ISRN INRIA/RR--8410--FR+ENG

A Scalable Communication Architecture for AMI in SmartGrid

Giang Ngo Hoang, Luigi Liquori, Hung Nguyen Chan

Project-Teams Lognet

Research Report n° 8410 — 03 December 2013 — 19 pages

Abstract: *Advanced Metering Infrastructure* (AMI), seen as foundation for overall grid modernization, is an integration of many technologies providing an intelligent connection between consumers and system operators [1]. One of the biggest challenges that AMI faces is to gather and process huge amounts of metering data. This requires a scalable communication architecture which provides adequate aggregation bandwidth while allowing to efficiently process metering data. In this paper, we introduce a novel communication architecture for AMI in which metering data is aggregated and processed in a tree-like topology. Through analysis we show that the architecture is scalable and resilient in presence of failure and partly self-organization. The experiments performed in large scale French Grid5000 platform [2] also shows the efficiency of communication of the proposed architecture.

Key-words: advanced metering infrastructure, peer-to-peer architecture, scalability, interconnecting, distributed hash table

RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Une Architecture de Communication Évolutive pour AMI en SmartGrid

Résumé : Ce document présente une architecture P2P pour Advanced Metering Infrastructure dans Smart Grid.

Mots-clés : avancé des infrastructures, de lárchitecture peer-to-peer comptage, évolutivité, inter-connexion, table de hachage distribuée

Contents

1 Introduction

In an Advanced Metering Infrastructure, smart meters measure and collect all the informations on energy consumption and power quality and send these informations to a *Metering Data Management System* (MDMS). When arrived in a MDMS, metering data are validated, processed, estimated, and then fed to the system side applications such as a *Consumer Information System* (CIS), an *Outage Management System* (OMS), or an *Enterprise Resource Planning* (ERP) to help optimize operations, economics and consumer service. By current standards, a few kilobytes of data are collected from each smart meter every 15 minutes [3, 4]. In addition, metering data can be collected on demand for various purpose such as billing inquiries, outage extent verification, and verification of restoration [5]. In a real, large scale systems, the amount of metering data become extremely large and many existing communication architectures are not sufficient enough to deal with the waves of metering data due to the limitation in bandwidth and in data storage.

In this report, we address the above challenge by introducing an hybrid architecture, merging P2P and client-server communications so allowing scalable data collection, aggregation, and management. This architecture comprises multiple points of data aggregation and processing, i.e. MDMSs, which are geographically distributed and hierarchically organized. The metering data are first collected, and then aggregated and transferred through multiple levels of MDMSs in a tree manner, thus reducing the throughput of data after each level. While MDMSs are organized in a P2P architecture to feature self-organization, scalability and resilience, the connections from MDMSs to smart meters follow a more standard client-server model in order to be compatible with existing collectors and smart meters. As such, the main contribution of this report is the introduction of a new hybrid communication architecture for AMI featuring scalability, resilience and self-organization of MDMS units.

Our report is organized as follows. Section 2 introduces the related work and our motivations. In Section 3, we describe the proposed AMI communication architecture and we investigate its characteristics. An implementation of the communication architecture is presented in Section 4 and it is evaluated by a large scale experiment on the Grid5000 platform in Section **??**. Finally, Section 6 draws conclusions and introduces possible future work.

2 Related work

Several communication architectures for AMI have been proposed in the last few years.

The traditional communication architecture [1] being one central *Operation Center* (OC) receiving metering data from all customers through local data concentrators. With this centralized architecture, all metering data go through MDMSs at one OC which then feeds the appropriate data into system side applications. This architecture makes the system simple and easy to manage. However, as the scale of system ups to large, this architecture does not scale-up because of the high possibility to create the data communication bottlenecks in the zones close to the OC, as pointed out in [6], and because the unfeasibility in data processing due to the large data load, as pointed out in [7].

In [6], Authors introduced a model with some similarities with *Content Delivery Network* (CDN) [8] in which a central MDMS connects to multiple distributed MDMSs; an algorithm to calculate the optimal deployment of distributed MDMSs in their model was showed. The distributed architecture allowed the aggregation of data thus solved the problem of non scalability in data collection.

In [9], Authors proposed an infrastructure based on *Group Communication* using an hybrid adaptive multicast over public networks. By relying on group communication over public networks, this approach reduced the cost of investment but the reliability of communication as well as the latency of data collection are not guaranteed.

In [10], Authors investigated the storage and monthly billing processing architecture. They compared storage techniques including the centralized relational database, the distributed relational database and the key-value distributed database storage. Their work focused on achieving the scalability in processing metering data rather than achieving the scalability in collecting the metering data.

In [11], Authors proposed a model of using cloud computing for smart grid data management. The advantages of computing was the ability to provide huge storage, powerful processing and communication. The work focused on providing sufficient resources for smart grid data management, without any investigation on the distribution of necessary resources, i.e. points of data collection and management.

The work of [6] share some of our motivations, namely it proposed a communication architecture for scalable data collection using a CDN like model for MDMSs network. The main difference w.r.t. our work is that *i*) we connect MDMSs via a P2P network providing more resiliency and fault tolerance and *ii*) we do not include data processing and integration algorithms.

3 AMI architecture

3.1 Scalable P2P architectures

Among P2P architectures, *Distributed Hash Table* (DHT) can be seen as the most scalable one. However, the flat structure of DHT is not suitable for the aggregation of metering data since they naturally need to be aggregated in a hierarchical fashion. Many tree-based overlays have been introduced such as [12, 14, 13]. These overlays are designed for supporting systems with a large number of peers whose roles are equal, i.e. a peer can stay at any position in the tree. Under the failure of certain peers, other peers are reorganized and can stay at any level in the tree. However, in an AMI, functions of MDMSs at different levels are not equal. The MDMSs at highest level, i.e. closest to smart meters, process raw data and send the summarized data to MDMSs at lower level and so on. Hierarchical DHT (HDHT) is a scalable P2P architecture allowing the localization of its peers. However, the peers in HDHT are equal in roles, just like the tree-based overlays. As a matter of fact, we introduce a new hierarchical architecture for AMI in which the levels of MDMSs keeps remain after reorganizing. In this architecture, a DHT is used to connect groups of MDMSs at the same level to take advantages of their self-organization and deterministic routing.

3.2 Structure

Figure 1 illustrates the proposed AMI communication architecture. The AMI includes multiple OCs which are geographically distributed and hierarchically organized. Each OC contains a MDMS for collecting and analyzing metering data from customers belonging to the area that the OC is responsible for. The MDMSs are equipped with analytical tools to interact with system-side applications which locate in that OC. As each OC has one MDMS, we use a "box" to present both an OC and the perspective MDMS. Applications in an OC are presented by a "can".

The level-1 MDMS is responsible for whole area where AMI covers. The whole area is divided into multiple sub-areas that each of which is covered by a level-2 MDMS. An area covered by a

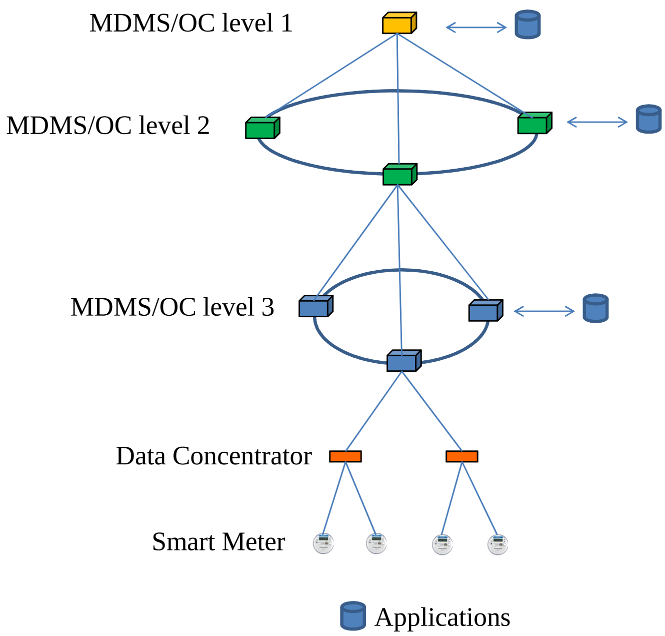

Figure 1: AMI proposed architecture

level-2 MDMS can be further divided into multiple sub-areas that each of which is covered by a level-3 MDMS and so on.

The level-n MDMS managing area S_n and the level-$(n+1)$ MDMS managing area S_{n+1} such that $S_{n+1} \subset S_n$ are called parent and child of each other respectively.

All MDMSs are connected in P2P architecture as following:

- All children of a level-n MDMS with $n \geq 1$ are organized in a DHT overlay network. The MDMS at level-n keeps contacts of all of its MDMSs children at level-$(n+1)$ while a MDMS at level-$(n+1)$ maintains the contact of its parent and contacts of parent's neighbors in the parent's overlay. In case that the parent MDMS is at level-1, its children maintain the contact of parent only.
- When the parent of a MDMS fails, the MDMS replaces its parent by the closest neighbor of the parent. If the old parent recovers from failure, the MDMS automatically changes the parent back to the old one. A MDMS also periodically check the aliveness of its children and remove the failed children from its children list.

Each MDMS at highest level manages a number of data concentrators following the client-server model. Each of the data concentrator collects metering data from a number of smart meters.

3.3 Data collection and processing

The processes of collecting and processing data is illustrated in the Figure 2. The information can travel in the system in three directions: upward direction, downward direction and intra-overlay.

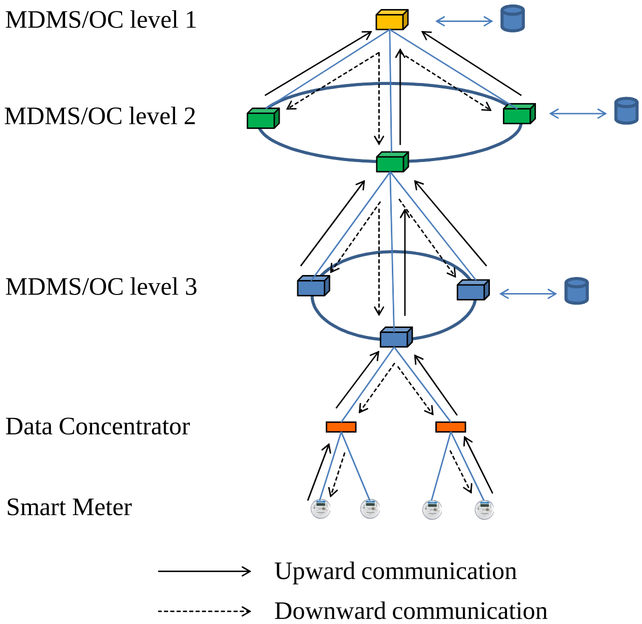

Figure 2: Communication in AMI

- Upward direction: the upward communication is illustrated by solid arrows in the Figure 2. In this direction, customers on-scheduled or on-demand send metering data to local concentrators which then send the data to the MDMSs managing them. At each MDMS, the data is processed and aggregated and the appropriate data is fed to local system-side applications, only the summarized data is sent to parent MDMS and so on.

- Downward direction: the downward communication is illustrated by dashed arrows in the Figure 2. In this direction, the MDMS at level-n can send control commands or data to MDMSs at level-$(n+1)$. The data and commands such as new energy price, request for reducing the load can also be send to customers through concentrators.

  There are three schemes of downward communication including broadcast, multicast or unicast. A MDMS at level-n can send the data or command to all of its children MDMSs or to a group of its children MDMSs or to one of its children MDMSs at level-$(n+1)$. A MDMS at highest level can send the data or command to all of its users or to a group of its users or even to a specific user.

- Intra-overlay: In this direction, the data and commands or other information can be exchanged between MDMSs in one overlay. The communication follows the protocol of that overlay.

System-side applications are deployed in OCs at appropriate levels depend on their functions, characteristics and requirements on latency. The applications that is sensitive with latency such as OMS, *Demand Response* (DR) should be deployed in OC at highest level, i.e. close to the smart meters, while other applications which are less sensitive with latency such as Billing system can be located in OC at lower level.

The distribution of applications in high level OCs also help to reduce the throughput of data flowing toward the lower level OCs thus make the system more scalable in term of data communication.

The MDMS and system-side applications co-located in one OC can interact with each other in several patterns. The MDMS can on-demand or on-scheduled feeds data to system-side applications. Other way of interaction exploits the Publish-Subscribe model [15] in which these applications subscribe for certain kind of event such as the outage flags generated by AMI system so that these applications are notified when the events happen. To enable these interactions, the utility can deployed analytical tools or publish-subscribed applications a long with the MDMS to provide the appropriate data to the applications.

We note that the information from system-side applications can be sent to customers via other paths such as Internet instead of AMI infrastructure. However, these communication paths are outside the scope of this work.

3.4 System analysis

This section investigates the characteristics of our architecture. We first define the meaning of some notation used in this report. The $\triangleq$ notation means “equal” while the $<<$ notation means “much less than”. The parameters of the model are denoted as following:

- M: the number of smart meters in the system;
- L: the largest level of OCs;
- λ_i: the number of messages generated by a smart meter in a period of time;
- α_n: the load over a MDMS at level n, calculated by a number of messages go to this MDMS. The load over a MDMS shows two aspects: the throughput of data flowing to the OC and the power processing needed to process these messages.
- β_n: the number of messages that a MDMS at level $n+1$ sent to its parent MDMS at level n in a period of time;
- $z_n \triangleq \dfrac{\beta_{n+1}}{\beta_n}$ with $1 \leq n \leq L-1$.

In case n=L, we let z_L be the number of messages a data concentrator sent to MDMS managing it over the number of messages that it receives from smart meters.

- K_n: the number of MDMSs in all OCs at level n;

We note that $z_n << 1$ because MDMS only sent summary data to its parent. This results in $K_n << K_{n-1}$.

Load scalability. According to [16], load scalability is the ability for a distributed system to easily expand and contract its resource pool to accommodate heavier or lighter loads or number of inputs. We define the load on AMI is the number of messages generated by all smart meters in one period of time which is calculated as following:

$$\sum_{i=1}^{M} \lambda_i \triangleq M \cdot \overline{\lambda}$$

with $\overline{\lambda}$ is the average traffic generated by a smart meter in a period of time. Thus the number of messages received by operation centers at level-n is:

$$M \cdot \overline{\lambda} \cdot \prod_{j=L}^{n} z_j$$

hence the load over a MDMS in level n is

$$\alpha_n \triangleq \frac{M \cdot \overline{\lambda} \cdot \prod_{j=L}^{n} z_j}{K_n} \tag{1}$$

with $n = \overline{1, L}$.

We note that in centralized model, the load on the central MDMS, denoted by α is:

$$\alpha \triangleq M \cdot \overline{\lambda} \tag{2}$$

From Formula 1 and Formula 2, we can see following points:

- By distributing the MDMSs and aggregating data at multiple levels, our model reduces the load on each OC comparing to the centralized model.
- By turning the number of aggregation levels and the number of OCs at each level, i.e. turning L and K_n, utility can control the load on each OC, i.e. control throughput and power processing requirement for each OC, thus eliminating the data transmission congestion as well as the overload in power processing.

  The adding or removing MDMSs to or from MDMS overlays can be achieved easily thanks to the self-organization advantages of P2P. If the change of load happens in a local area, utility only need to adapt the number of MDMSs in the OCs which are responsible for that area.

  The easiness of changing the resource pool to adapt with either the global changes or the local changes of load shows the load scalability of AMI based on our architecture.

3.4.1 Failure resilience

The proposed communication architecture exploits the resilience with failure advantage of P2P in managing the MDMS network.

According to [17], resilience means the capacity of a system to withstand sudden, unexpected failures, and (ideally) to be capable of recovering quickly afterwards. Resilience implies both strength and flexibility in the sense that a resilient structure would bend, but would be hard to break.

In our proposed architecture, the data of a MDMS can be replicated at its neighbors in the DHT overlay. As the MDMS fails, the closest neighbor automatically replaces it. On the other hand, a MDMS normally sends the data to its parent MDMS. In case the parent MDMS fails, the MDMS automatically send the data to the closest neighbor of the parent MDMS.

Similarly, a data concentrator normally sends the data to its parent. In case the parent fails, it will send the data to closest neighbor of the parent. To deal with the situation that the level-1 OC fails, utility can deploy P2P applications for retrieving, querying data over overlay of MDMS at level-2.

As a matter of fact, even if the failure happen in some MDMSs, the system will quickly recovers and then works normally.

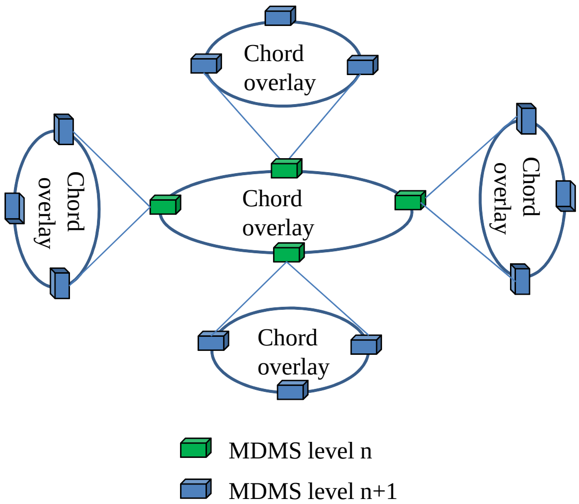

Figure 3: Evaluation communication architecture

3.4.2 Low cost of maintenance and operation

With the large scale and the growth of AMI infrastructure, the self-organization ability of the MDMS network is very important in the sense that it helps the utility reduces the operation and maintenance cost. When the failures happen, the network automatically recovers without the human intervention. As the scale of system change, utility can easily add or remove MDMSs with little early configuration.

4 An implementation of communication architecture

We implemented and evaluated the fundamental part of the communication architecture in which one MDMS overlay at level-n connects to multiple MDMS overlays at level-$(n+1)$. We extended the Chord protocol [18], a typical DHT, to implement these MDMS overlays as a case study. Other DHTs can also be used instead.

The other parts, namely the MDMS level-1 connect to overlay of level-2 MDMS and a MDMS connects to data concentrators in client-server architecture, are well investigated architectures. Therefore we did not implement and evaluate these parts. The implemented architecture is illustrated in Figure 3.

4.1 ID assignment

We assign an unique p-bits identifier to each MDMS as following:

- The level-1 MDMS is assigned a random p-bits number;
- Each of the level-2 MDMSs is assigned a p-bits number in which q first bits is random and $p-q$ last bit is set to 0;

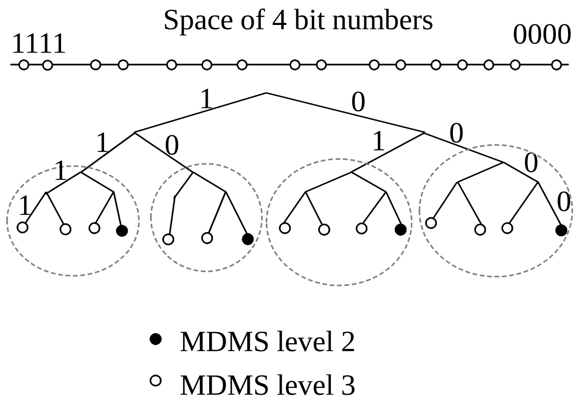

Figure 4: ID assignment example

- Assume that a level-n MDMS with $n \geq 2$ has identifier with first $(n-1) \cdot q$ bits is m_0. Its MDMSs children, i.e. the MDMSs in OC at level-$(n+1)$ has identifiers specified as following: first $(n-1) \cdot q$ bits is m_0, next q bits is random and $p - n \cdot q$ last bits is set to 0.

Figure 4 illustrates a simple example of ID assignment. The inter-connecting DHT structure includes three levels (see Figure 2) in which one level-1 MDMS connects to an overlay network of level-2 MDMSs. Each of the level-2 MDMS connects to an overlay network of level-3 MDMSs. For the sake of simplicity, in the example, the identifiers of MDMSs are presented by only 4 bits numbers, i.e. $p = 4$, here we set $q = 2$.

Note that the identifier of level-1 MDMS is arbitrary. The identifiers of level-2 MDMSs are presented by black bullets with two last bit is set to 0. Each of the level-2 MDMS has several children, i.e. level-3 MDMSs, whose identifiers are represented by plain circles and share the first two bits with the level-2 MDMS. Therefore, a level-2 MDMS and its children belong to the same subtree bounded by a dashed circle.

4.2 Parent-children relationship

4.2.1 Parent-children assignments

A MDMS `X` peer in the child overlay is the child of a MDMS `Y` in the parent overlay if identifier of `X` stays between identifier of `Y` and identifier of `Y`'s successor, i.e. the closest succeeding node of `Y` in the identifier space.

Let `X@Y` denotes `X` is child of `Y`. Assume that `Z` is successor of Y, and $\mathtt{ID_X}$, $\mathtt{ID_Y}$ and $\mathtt{ID_Z}$ be the identifiers of `X`, `Y` and `Z`, respectively. Then we have that:

$$\mathtt{X@Y} \; \textit{if} \; \mathtt{ID_X} \in [\mathtt{ID_Y}, \mathtt{ID_Z})$$

Figure 5 illustrates an example in which the parent MDMS overlay, i.e. O1, connects to three children MDMS overlays, namely O2, O3 and O4. In the figure, MDMSs are denoted by black bullets. A, B and C are MDMSs in O1. C is successor of B which is successor of A. D, E and F are MDMSs in O2 and are children of A because $\mathtt{ID_D}$, $\mathtt{ID_E}$ and $\mathtt{ID_F}$ belong to $[\mathtt{ID_A}, \mathtt{ID_B})$. Similarly, G, H and I are MDMSs in O3 and are children of C while K, M and N are MDMSs in O4 and are children of B.

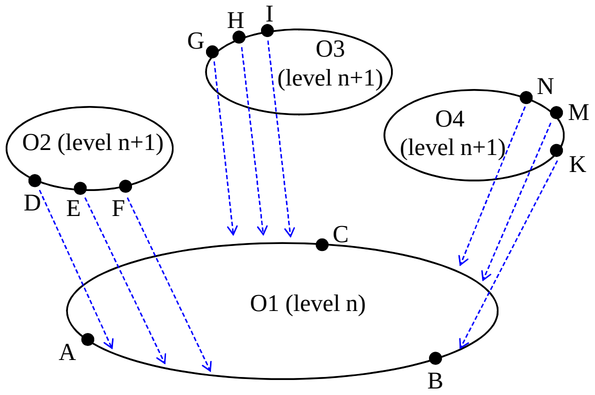

Figure 5: Parent children example

4.2.2 Children list and parent list

A peer maintains a children list and a parent list to keep the information of its children and parents. The children list of a peer contains information about all of its children. The parent list contains the information about the parent peer and parent peer's predecessor peers, i.e. the closest preceding peers of the peer in the Chord identifier space.

4.3 Communication schemes

Utility can employ various communication schemes to support communication among MDMSs. These schemes can be categorized into three categories based on the direction of the communication: intra-overlay communication, downward communication and upward communication.

4.3.1 Intra-overlay communication

is the communication between MDMSs in one DHT overlay. A MDMS can employs unicast, broadcast over DHT and multicast over DHT for intra-overlay communication. In this report, we called these schemes intra-unicast, intra-multicast and intra-broadcast to distinguish them with similar schemes in downward communication.

Many studies such as [19, 20] proposed algorithms for broadcast over a DHT. In our implementation, we used the broadcast algorithm proposed by [19] as intra-broadcast while the intra-multicast algorithm is adopted from [19] by ourselves.

4.3.2 Downward communication

is the communication from parent MDMSs to children MDMSs. A MDMS can use any of the three schemes: downward-unicast, downward-multicast and downward-broadcast to respectively send information, i.e. data or commands, to its specific child MDMS or to a group of its children MDMSs or to all of its children MDMSs. For example, in the Figure 5, MDMS A can send information to a specific MDMS among D, E and F or to a group of or to all of them.

4.3.3 Upward communication

is the communication from a MDMS to its parent MDMS. A MDMS use upward-unicast to send information to its parent MDMS. For example, in the Figure 5, MDMSs K, M and N can send information to B which is their parent MDMS.

4.3.4 Combination scheme

is the combination between intra-overlay communication and downward communication: it allows the utility to unicast, multicast or broadcast information, i.e. data or command, to a specific MDMS, a group of MDMSs or to all MDMSs in any area. The idea of combination scheme is as following. The sending MDMS first employs intra-unicast or intra-multicast or intra-broadcast to send the information to a specific MDMS or to a group of MDMSs or to all MDMSs in the same overlay with it. The receiving MDMS then forwards the information to one of its children or to group of its children or to all of its children. The combination scheme constitutes three kind of communication as following:

- Combination-unicast: used by a MDMS at level-n to send information of a specific MDMS at level-$(n+1)$. For example, in Figure 5, MDMS A can send information to a specific MDMS belonging to any of three overlays: O2, O3 and O4.
- Combination-multicast: used by a MDMS at level-n to send information of a group of MDMS at level-$(n+1)$. For example, in Figure 5, MDMS A can send information to a group of MDMSs belonging to any of three overlays: O2, O3 and O4.
- Combination-multicast: used by a MDMS at level-n to send information of all MDMSs at level-$(n+1)$. For example, in Figure 5, MDMS A can send information to all of the MDMSs in three overlays: O2, O3 and O4.

5 Evaluation

In this section, we evaluate the communication efficiency, the maintenance traffic and the scalability of the proposed communication architecture via evaluating the implemented one.

5.1 Objectives

This section evaluates following characteristics of the proposed P2P architecture: *i*) The efficiency of communication in term of the ratio of successful communication; *ii*) The traffic for maintaining the network of MDMSs characterized by the amount of traffic generated by a MDMS in a period of time; *iii*) The scalability of the communication architecture.

The intra-overlay communication was evaluated in many previous studies such as [18, 19]. Therefore we focus on evaluating the efficiency of combination scheme and upward communication including: combination-unicast, combination-multicast and upward-unicast.

In this report, the scalability of the communication architecture is evaluated through the change of communication efficiency and the change of communication cost, i.e. generated traffic, as the scale of the system increase. In the experiments, the scale of the P2P system increased up to 4 times: the number of MDMSs in parent overlay increased from 10 to 40 while the number of children overlays increases from 10 to 40, i.e. the number of children MDMSs increased from 500 to 2000.

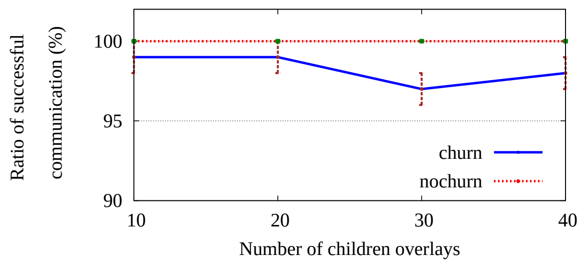

Figure 6: Success ratio in downward-multicast

5.2 Experiment setup

5.2.1 Evaluation architecture

the evaluation architecture is illustrated in Figure 3. In the experiment, a P2P system with one level-2 MDMS overlay connects to various numbers of MDMS overlays at level-3 has been deployed on the large scale French Grid5000 platform [2].

5.2.2 Evaluation scenario

we assume that a data concentrator receive the metering data from 100 smart meters while each MDMS receive the data from 100 data concentrators on average. Thus a MDMS manages $100 \cdot 100 = 10000$ smart meters on average. We evaluate the communication architecture for large scale AMI with the number of smart meter from 5 millions to 20 millions. In this scenario, the number of MDMS at highest level (level-3) varies from $5000000/10000 = 500$ to $20000000/10000 = 2000$. We also assume that each MDMS level-2 manages 50 MDMSs level-3, thus the number of MDMS at level-2 varies from $500/50 = 10$ to $2000/50 = 40$. This also means that the number of children overlay varies from 10 to 40. The experiments are performed in both no churn and high churn environments to show the efficiency of communication architecture in ideal condition and the resilience of the communication architecture under the failure of MDMSs respectively. In churn condition, lifetime mean of parent peers is set to 2 hours while lifetime mean of children peers is set to 1 hours. Each experiment is run 5 times. The average values and standard deviation of evaluated metrics are plotted in the figures in following sections. The values of experiment parameters are described in Table 1.

No. of parent overlay	1
No. of children overlay	10, 20, 30, 40
No. of node per child overlay	50
No. of parent nodes	10, 20, 30, 40
Churn	high churn, no churn

Table 1: Values of experiment parameters

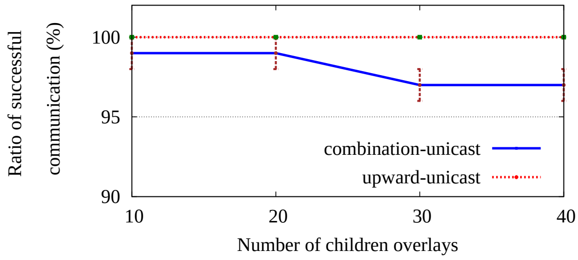

Figure 7: Ratio of successful communication in unicast schemes

5.3 Experiment results: Communication efficiency and Scalability

This section shows the experimental results on the ratio of successful communication in combination scheme and upward communications.

5.3.1 Combination-multicast

the ratio of successful communication performed in combination-multicast scheme is illustrated in Figure 6. The lines "churn" and "nochurn" represent ratio of successful communication, performed in the systems under churn condition and no churn condition respectively, when the number of children overlays increases from 10 to 40. The figure also shows that the combination-multicast communication performed in the no churn environment succeed with the ratio of 100%. In the churn environment, the value of this ratio slightly changes in the range from 99% to 97% as the number of children overlay increases from 10 to 40 and the standard deviations are less than 1.5 for all cases.

Let $\Delta_{com\_multicast\_churn}$ and $\Delta_{com\_multicast\_nochurn}$ be the percentage change of successful communication ratio in combination-multicast as the scale of the system increases in the churn and no churn environment respectively. Then:

$$\Delta_{com\_multicast\_nochurn} = \frac{100 - 100}{100} = 0\%$$

and

$$\Delta_{com\_multicast\_churn} = \frac{99 - 97}{99} = 2\%$$

5.3.2 Combination-unicast and upward-unicast

Figure 7 illustrates the ratio of successful communication in combination-unicast and upward-unicast schemes. The two lines "combination-unicast" and "upward-unicast" represent the ratio of successful combination-unicast communication and upward-unicast communication respectively, performed in the systems which are under high churn condition, as the number of children overlay rises from 10 to 40. From the figure we can see that the line "combination-unicast" is slightly changes in the range from 97% to 99% with the increase of number of children overlays from 10 to 40. The standard deviations are less than 1.5 for all cases. On the other hand, the line "upward-unicast" is horizontal line at value of 1. We did not show the ratio of successful communication in combination-unicast and upward-unicast in no churn condition which are 1 for all cases for the sake of clarity.

Let $\Delta_{com\_unicast\_churn}$ and $\Delta_{com\_unicast\_nochurn}$ be the percentage change of successful communication ratio in combination-unicast as the scale of the system increases in the churn and no churn environment respectively. Then:

$$\Delta_{com\_unicast\_nochurn} = \frac{100 - 100}{100} = 0\%$$

and

$$\Delta_{com\_unicast\_churn} = \frac{99 - 97}{99} = 2\%$$

Let $\Delta_{up\_unicast\_churn}$ and $\Delta_{up\_unicast\_nochurn}$ be the percentage change of successful communication ratio in upward-unicast as the scale of the system increases in the churn and no churn environment respectively. Then:

$$\Delta_{up\_unicast\_churn} = \Delta_{up\_unicast\_nochurn} = \frac{100 - 100}{100} = 0\%$$

The percentage changes in communication efficiency in various communication schemes as the scale of the system increase up to 4 times is summarized in the Table 2. The Δ notation in the table means the percentage change in communication efficiency.

5.3.3 Discussion and Analysis

the high values of ratio of successful communication in both high churn and no churn conditions along with the low standard deviation values show the efficiency and the stable of all evaluated communication schemes: combination-multicast, combination-unicast and upward-unicast. On the other hand, the slightly changes or even no changes in ratio of successful communication in evaluated communication schemes (see Table 2) as the scale of the system increase 4 times, i.e. the number of parent MDMSs increase from 10 to 40 while the number of children MDMS increase from 500 to 2000, shows the scalability of the communication architecture.

5.4 Experiment results: Maintenance traffic and Scalability

In this report, we extend Chord DHT to implement MDMS overlays. Therefore, in this section, we evaluate the maintenance traffic of the proposed architecture by comparing the traffic generated by a parent peer and a children peer with the traffic generated by a Chord peer in the same condition. In Figure 8, the two lines "parent vs. chord" and "child vs. chord" show the ratio of the traffic generated by a parent peer and a child peer respectively over the traffic generated a Chord peer in the same size network in churn condition.
From Figure 8, we can see that the line "child vs. chord" is horizontal at the value of 1.1. The line "parent vs. chord" is slightly decreases from 2.8 to 2.3 as the number of children overlay increase from 10 to 40. The standard deviations are approximately 0 in all cases.

Communication schemes	Δ no churn	Δ churn
Combination-multicast	0%	2%
Combination-unicast	0%	2%
Upward-unicast	0%	0%

Table 2: Percentage changes in communication efficiency in various communication schemes

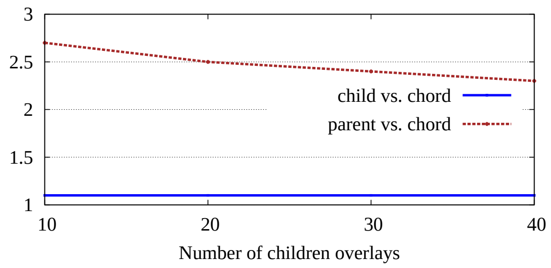

Figure 8: Average traffic generated by parents and children peers

5.4.1 Discussion and Analysis

the fact that the traffic generated by a child peer based on Chord equals 1.1 times the traffic generated by a Chord peer in the same size network, shows that additional traffic generated in a child peer is very small. We note that the maintenance traffic of a peer, running DHT protocol, increases when the number of peers in the DHT overlay increases. In our experiment, when the number of children overlay increases from 10 to 40, i.e. the number of parent peer also increases, the line "parent vs. chord" is slightly decrease. This means that the maintenance traffic of parent overlay increases with slower rate than the maintenance traffic of Chord overlay as the size of the overlay increases. This proves the scalability of parent overlay in term of maintenance traffic.

5.4.2 Experiment results: Summary

the experiments shows that our proposed communication architecture achieves *high communication efficiency* and *low maintenance traffic* for maintaining children overlays. The architecture also features by the *scalability* in term of both the communication efficiency and the maintenance traffic.

6 Summary

In this report, we have introduced a new AMI communication architecture for scalable data collection and management. The architecture is mixing of P2P and client-server model in which the MDMSs are geographically distributed and hierarchically organized in a P2P manner. The MDMSs at highest level play the roles of server managing data concentrators.

The analysis shows that the AMI based on our communication architecture is scalable for data collection and management while being resilient with failure and allow reducing the operation cost. Utility can also plan the geographic distribution of MDMSs to have expected latency of collecting data. Smart Grid applications can be deployed in OCs at different levels depending on their characteristics and requirements on latency.

The experiments shows the efficiency of communication schemes in the proposed architecture in term of ratio of successful communication in both high churn and no churn environments. The communication is performed with at least 97% of success under the high churn and with 100% of success in no churn condition. The experiments also show that the children peers generate low maintenance traffic, i.e. 1.1 times larger than a Chord peer in the same size network. The scalability of the communication architecture is shown in both aspects: communication efficiency and maintenance traffic.

References

[1] US Dept. of Energy, Office of Electricity Delivery and Energy Reliability. *Advanced Metering Infrastructure.* 2008

[2] *Grid5000 platform.* Online at `https://www.grid5000.fr/mediawiki/index.php/Grid5000:Home`. Accessed June 2013.

[3] *SmartGrid/AEIC AMI interoperability standard guidelines for ANSI C12.19 / IEEE 1377 / MC12.19 end device communications and supporting enterprise devices, networks and related accessories.* The Association of Edison Illuminating Companies, Meter and Service Technical Committee report version 2. 2010.

[4] D. Bernaudo et al. *SmartGrid/AEIC AMI interoperability standard guidelines for ANSI C12.19 / IEEE 1377 / MC12.19 end device communications and supporting enterprise devices, networks and related accessories.* Online at `http://www.aeic.org/meterservice/AEICSmartGridStandardv2-11-19-10.pdf`. The Association of Edison Illuminating Companies, Meter and Service Technical Committee report version 2. 2010.

[5] R. H. Khan and J. Y. Khan. *A comprehensive review of the application characteristics and traffic requirements of a smart grid communications network.* Computer Networks, 2013.

[6] Z. Jiazhen and R.Q. Hu and Y. Qian. *Scalable Distributed Communication Architectures to Support Advanced Metering Infrastructure in Smart Grid.* IEEE Transactions on Parallel and Distributed Systems. 2012.

[7] C. Gerdes and U. Bartlang, and J. Muller. *Vertical Information Integration for Cross Enterprise Business Processes in the Energy Domain.* Proceedings of Agent-Based Technologies and Applications for Enterprise Interoperability. 2009.

[8] Wikipedia. *Content Delivery Network.* Online at `http://en.wikipedia.org/wiki/Content_delivery_network`. Accessed June 2013.

[9] S. Meiling and T. Steinbach and T. C. Schmidt and M. Wählisch. *A Scalable Communication Infrastructure for Smart Grid Applications using Multicast over Public Networks.* Proc. of ACM SAC. 2013.

[10] M. Arenas-Martinez and S. Herrero-Lopez and A. Sanchez and J.R. Williams and P. Roth and P. Hofmann and A. Zeier. *A Comparative Study of Data Storage and Processing Architectures for the Smart Grid.* Proc. of SmartGridComm. 2010.

[11] S. Rusitschka and K. Eger and C. Gerdes. *Smart Grid Data Cloud: A Model for Utilizing Cloud Computing in the Smart Grid Domain.* Proceedings of IEEE First Int'l Conf. Smart Grid Comm, 2010.

[12] H. V. Jagadish and B. C. Ooi and K.L. Tan and Q. H. Vu and R. Zhang. *BATON: a balanced tree structure for peer-to-peer networks.* Proceedings of the 31st international conference on Very large data bases. 2005.

[13] J. Leitao and J. Pereira and L. Rodrigues. *Epidemic Broadcast Trees.* Proceedings of the 26th IEEE International Symposium on Reliable Distributed Systems. 2007.

[14] M. Li and W.c. Lee and A. Sivasubramaniam. *DPTree: A Balanced Tree Based Indexing Framework for Peer-to-Peer Systems.* Proceedings of the IEEE International Conference on Network Protocols. 2006.

[15] *Publish Subscribe pattern.* Online at http://en.wikipedia.org/wiki/Publish%E2%80%93subscribe_pattern. Accessed Oct 2013.

[16] Wikipedia. *Scalability.* Online at http://en.wikipedia.org/wiki/Scalability. Accessed June 2013.

[17] J. Cascio *Resilience.* Online at http://www.fastcompany.com/blog/jamais-cascio/open-future/resilience. Accessed September 2013.

[18] I. Stoica and R. Morris and D. Karger and M.F. Kaashoek and H. Balakrishnan. *Chord: A scalable peer-to-peer lookup service for internet applications.* SIGCOMM, 2001

[19] S. El-Ansary and L. Alima and P. Brand and S. Haridi. *Efficient Broadcast in Structured P2P Networks.* IPTPS 2003

[20] M. Castro and P. Druschel and A-M . Kermarrec and A. I.T. Rowstron. *Scribe: a large-scale and decentralized application-level multicast infrastructure.* IEEE J.Sel. A. Commun. 2006

RESEARCH CENTRE
SOPHIA ANTIPOLIS – MÉDITERRANÉE

2004 route des Lucioles - BP 93
06902 Sophia Antipolis Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399